EXAMINER'S AMENDMENT

1. An examiner's amendment to the record appears below. Should the changes and/or additions be unacceptable to applicant, an amendment may be filed as provided by 37 CFR 1.312. To ensure consideration of such an amendment, it MUST be submitted no later than the payment of the issue fee.

2. Authorization for this examiner's amendment was given in a telephone interview with Aly Z. Dossa (Reg. No. 63,372), on December 17, 2008, for obviating any potential 35 U.S.C. 112/2nd issues, and to put the claims in condition for allowance.

3. The application has been amended as follows:

Please amend the claims as indicated in *complete listing* of the claims attached to the end of this document. – *See pages 5-7.*

Allowable Subject Matter

4. Claims 1-4, 7-11, and 13 are allowed.

5. The following in an Examiner's Statement of Reasons for allowance:

The prior Art of record, taken alone or in combination do not disclose and/or suggest the limitation as recites in per independent claims 1 and 8, at least, as:

"obtaining an original instruction associated with the probe, wherein obtaining the original instruction comprise searching a look-up table using a program counter value, wherein the look-up table comprises the original instruction associated with the probe and an address associated with the original instruction, and wherein the program counter value corresponds to a current address of a first thread executing in the instrumented program" and "loading the original instruction into the second scratch space, wherein the scratch space is allocated on a per-thread basis, and wherein a first scratch space for the first thread executing in the instrumented program was previously allocated", as reflected in Applicants' arguments (*See Remarks, page7: last paragraph, and page 8, ¶ 1 & 2)*.

The limitation as indicated above are not presented in the prior Arts and would not have been obvious. Thus, claims 1-4, 7-11, and 13 are in condition for allowance.

6. Any comments considered necessary by applicant must be submitted no later than the payment of the issue fee and, to avoid processing delays, should preferably accompany the issue fee. Such submissions should be clearly labeled "Comments on Statement of Reasons for Allowance".

Conclusion

7. Any inquiry concerning this communication or earlier communications from the examiner should be directed to MARINA LEE whose telephone number is (571)270-1648. The examiner can normally be reached on M-F (11:00 am to 7: 30 pm) EST.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Q. Dam can be reached on (571) 272-3695. The fax phone number for the organization where this application or proceeding is assigned is 571-273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free). If you would like assistance from a USPTO Customer Service Representative or access to the automated information system, call 800-786-9199 (IN USA OR CANADA) or 571-272-1000.

/M. L./
Examiner, Art Unit 2192

/Tuan Q. Dam/
Supervisory Patent Examiner, Art Unit 2192

EXAMINER'S AMENDMENT

1. (Currently Amended) A method for tracing an instrumented program on a processor having an x86 architecture, comprising:

triggering a probe in the instrumented program;

obtaining an original instruction associated with the probe, wherein obtaining the original instruction comprises searching a look-up table using a program counter value, wherein the look-up table comprises the original instruction associated with the probe and an address associated with the original instruction[[;]], and wherein the program counter value corresponds to a current address of a first thread executing in the instrumented program;

allocating a second scratch space for a second thread;

loading the original instruction into the second scratch space, wherein the scratch space is allocated on a per-thread basis, and wherein a first scratch space for [[a]] the first thread executing in the instrumented program was previously allocated;

loading a jump instruction for the x86 architecture into the second scratch space wherein the jump instruction includes a next program counter value;

executing the original instruction in the second scratch space using [[a]] the second thread to collect data; and

executing the jump instruction in the second scratch space using the second thread.

2. (Currently Amended) The method of claim 1, further comprising:

emulating the original instruction to determine the program counter value if the original instruction is a control-flow instruction; and

returning control to the second thread at an address of the program counter value if the original instruction is [[a]] the control-flow instruction.

3. (Previously Presented) The method of claim 1, further comprising:

determining the next program counter value by incrementing the program counter value using a size of the original instruction.

4. (Original) The method of claim 1, wherein the probe corresponds to a trap.

5. (Canceled)

6. (Canceled)

7. (Original) The method of claim 1, wherein the instrumented program is executed on a multi-thread architecture.

8. (Currently Amended) A system for tracing an instrumented program on a processor having an x86 architecture, comprising:

a first thread configured to execute the instrumented program;

a second thread configured to execute the instrumented program;

a first scratch space allocated for the first thread;

a program counter value corresponding to a current address of the first thread;

a look-up table arranged to store an address and a corresponding original instruction;

a trap handler configured to halt execution of the second thread when a trap instruction corresponding to a probe is encountered, use ~~an address of the trap instruction~~ the program counter value to obtain the corresponding original instruction from the look-up table, and ~~generate~~ load a jump instruction into a second scratch space ~~to an address in the instrumented program~~, wherein the jump instruction includes a next program counter value;

~~a first scratch space allocated for the first thread;~~

[[a]] the second scratch space arranged to store the original instruction and the jump instruction, wherein the second scratch space is allocated on a per-thread basis, and wherein the first scratch space for the first thread executing in the instrumented program was previously allocated; and

an execution facility for executing the original instruction in the second scratch space to collect data and executing the jump instruction, wherein the execution facility is a processor based on the x86 architecture.

9. (Original) The system of claim 8, further comprising:

a buffer for storing the data.

10. (Currently Amended) The system of claim 8, further comprising:

a tracing framework configured to emulate the original instruction to determine a value of [[a]] the program counter if the original instruction is a control-flow instruction and to return control to a thread at an address of the program counter value if the original instruction is [[a]] the control-flow instruction.

11. (Original) The system to claim 8, wherein the trap handler sets a destination of the jump instruction to a next address immediately following an address of the trap instruction.

12. (Canceled)

13. (Original) The system of claim 8, wherein the instrumented program is executed on multi-thread architecture.